LS-185
O. D. Despe
April 15, 1990

# Power Supply Control Units for APS Ring Magnets

## Introduction -

The APS storage ring (1104 meters) is divided into 40 sectors. Each sector has 38 magnet coils in five magnet bases. Every alternate sector has an additional quadrupole magnet for skew correction. All the main dipole magnets, two in each sector are connected in series and fed from one power supply unit. A base is controlled by one power supply control unit (PSCU). Each PSCU is connected to the host computer via a local area network (LAN). This note discusses the hardware configuration of the typical power supply control system used by the APS magnets and the software commands supported by the PSCU.

## 1.0 Power Supply Control Unit -

### 1.1 Hardware

The PSCU is a 68000 microprocessor-based system housed in a 16-slot Eurocard chassis using a G-64/G-96 compatible bus. A 200 watt built-in power supply provides ±12 V and 5V power to the chassis. The compatible commercial cards (manufactured by GESPAC Inc. of Mesa, AZ) or functional equivalents for a typical base configuration are:

| Items | # of Cards |
|---|---|
| Processor Card | 1 |
| RAM Card | 1 |
| Opto Coupled Serial I/O | 1 |
| ADC with differential inputs | 2 (32 Inputs) |
| Opto-coupled digital Inputs | 2 (64 bits) |
| Opto-coupled digital outputs | 2 (64 bits) |
| Timers 16-bit * | 2 (20) |
| Precision ADC Interface * | 1 |

{ * The 16-bit timer card (10 timers to a card) and the precision ADC interface card are custom designs.}

## Processor Card (GESMPU-4B)

This is an 8 MHz Motorola 68000 cpu-based card with Microware's OS-9 operating system. It has a built-in RS-232 serial port, four 28-pin sockets for system and application RAM/ROM memory. Three programmable timers are also available.

## RAM Card (GESMEM-12)

Gesmem-12 is a universal Ram/EPROM module with capacity to 512 Kbytes in eight 28-pin sockets.

## Serial I/O

At this time, the RS-232 port in the CPU board is used for communicating with the host. It is anticipated that some other network hardware and protocol that provides higher speed and data transmission integrity will be used in the final version of the PSCU.

## Differential ADC (GESADC-3)

This unit provides 16 channels of differential inputs. It is built around Analog Devices AD576 (12-bit ADC) and a programmable gain amplifier. Its input range is limited to +/- 1.0 V. Its conversion time is 35 us.

## Digital Inputs (GESINP-2A)

32 optically coupled digital signals are connected to this unit via two 26-pin connectors. The input level required is 12 V at 1.6KΩ. Isolation is rated at 1500V.

## Digital Outputs (GESOUT-3A)

32 optically coupled digital signals are output from this unit via two 26-pin connectors. The open collector output lines can drive up to 100 mA. Isolation is rated at 1500V.

## Timer Card (Custom Design)

Ten programmable 16-bit timers ( two 9513 counter chips) together with an 8-bit digital I/O for control and monitoring are provided. This card generates a programmed number of pulses for 4 power supplies. Each power supply can be programmed either to generate the pulses upon receiving the command (SyncMode=0), or wait for an external trigger (SyncMode=1), before delivering the pulses. It also provides the main clock pulse train used by the pulse width modulated (PWM) chopper supplies. Its outputs are differential line drivers for optical couplers to provide isolation and reduced interference pickup.

## ADC Interface (Custom Design)

This unit accepts pulse train inputs from up to 8 remote ADCs. This allows the monitoring of analog signals of higher precision provided by the GESADC-3 described earlier. The serial data inputs are all optically coupled for isolation and reduced interference pickup.

### 1.2 Software

The embedded application software that runs in the PSCU is written in the C-language under Microware's OS-9 operating system. The main modules that make up the application are the Constants and Variables Module, Devices module, Timers module, Command and Communication module, Execution module and Main module.

All these modules when compiled and linked together, produce the executable code that is burned into the application EPROMs. All the commands in the PSCU vocabulary are listed and described in later sections of this design note.
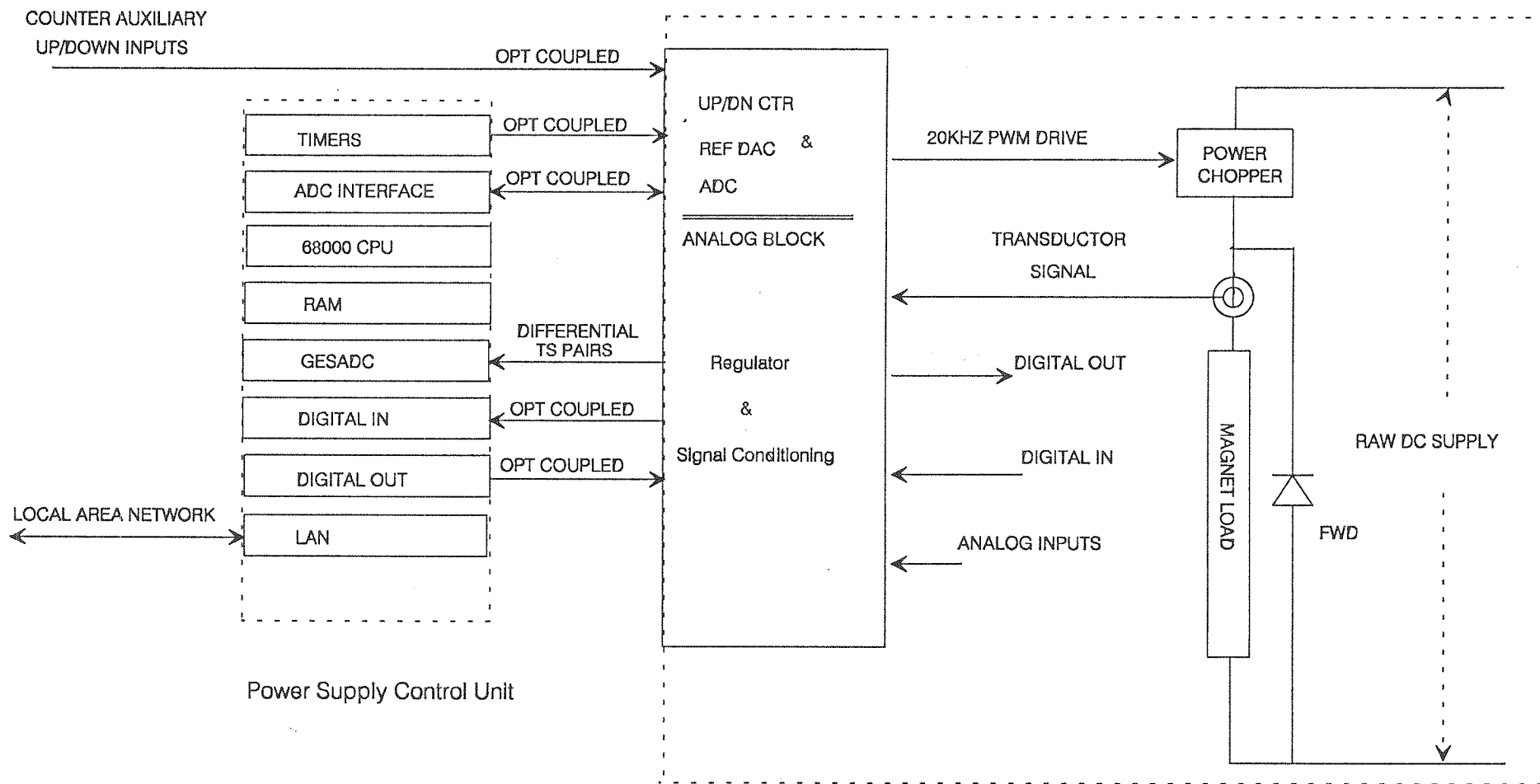
### 2.0 General Operation -

The typical magnet power supply control chain starts from the VAX or VME host, to the power supply control unit, to the regulator and signal conditioning block down to the driver/power chopper unit (See Fig. 1.) This chain is bidirectional to allow the required send/receive functions between the blocks. The PSCU communicates to each power supply using optically coupled digital signals and differentially shielded twisted pairs for analog signal transmission.

The command packet transmitted to the PSCU from the host contains the node address (cabinet number), the command code, power supply number ID and the required parameter bytes appropriate for the command, in addition to the bytes required for the host communication protocol. The node address selects one of the 200 cabinet racks on top of the storage ring tunnel. Each rack has one PSCU that handles the operation and monitoring of six to eight magnet/power supply pairs.

When a command is received from the host via the LAN, the PSCU receives an interrupt from the communication hardware. The message which is sent according to a specific format, is decoded into its various components (command, power supply #, and appropriate parameter values) which the embedded PSCU software uses to execute the command. All commands received are compared to a table of commands supported by the particular PSCU; an invalid command results in a null operation. See Table 1.

In the absence of a command from the host, the PSCU software executes a continuous loop of monitoring for host messages, continual update of its power supply statuses and data buffers and checking for alarm condition and reporting to host if alarm condition exists. Thus the data in each power supply database is virtually live. This

FIG. 1 - POWER SUPPLY CONTROL SYSTEM

COUNTER AUXILIARY
UP/DOWN INPUTS

OPT COUPLED

UP/DN CTR

REF DAC    &

ADC

ANALOG BLOCK

TIMERS          OPT COUPLED

ADC INTERFACE   OPT COUPLED

68000 CPU

RAM

GESADC          DIFFERENTIAL
                TS PAIRS

DIGITAL IN      OPT COUPLED

DIGITAL OUT     OPT COUPLED

LOCAL AREA NETWORK

LAN

Power Supply Control Unit

Regulator

&

Signal Conditioning

20KHZ PWM DRIVE

POWER
CHOPPER

TRANSDUCTOR
SIGNAL

DIGITAL OUT

DIGITAL IN

ANALOG INPUTS

MAGNET LOAD

RAW DC SUPPLY

FWD

Typical Regulated Power Supply with Load

software loop executes in about 32 ms when no analog channel is actively monitored on the display, or 130 ms when a channel is actively displayed.

Upon power up, the default modes and values for the relevant variables in the PS database are as follows:

|  |  |  |
|---|---|---|
| SyncMode=0 | | Normal mode |
| Ramp Rate Generator frequency | | 20kHz |
| DAC range | | 0 .. 65535 |
| Scale Factors | | |
| | Magnet current | TBD |
| | Magnet voltage | TBD |
| | DAC output | TBD |
| | Heat Sink temp | TBD |
| | Water temp | TBD |
| Unspecified Variables | | 0 |

## 2.1 Analog Reference Block

The Analog Reference Block is a remote subsytem to the PSCU and is located generally in the regulator chassis ( See Figure 1). It is the key subsystem that determines how the output of the regulated power supply is controlled. The components in this block are the UP/DOWN counter, the reference DAC and the compatible ADC. In order to simplify the control of the Analog Reference Block and provide isolation and reduced noise interference pickup, optically coupled pulses from the Timer card are supplied to the UP/DOWN counter that sets the reference DAC.

DACs with three different resolutions are used: 18 bits, 16 bits and 14 bits. The slowest conversion time is at least 40 usecs to full scale. Optically coupled auxiliary inputs to UP/DOWN counters are also provided to permit external dynamic modification of the reference DAC analog output.

The ADC with a resolution higher than or equal to its associated DAC, located on the same printed circuit board as the reference DAC in each power supply regulator unit, is fed the transductor signal (0 to 10V FS). It converts this signal to a binary value. The 18-and 16-bit ADCs are constantly acquiring and updating data in their internal memory independent of external control at 4 kHz rate; however, its digital filter has an equivalent 6-pole gaussian response corner frequency of 10 Hz. In cases where it is necessary for the ADC output to accurately track its analog input at higher frequencies, an appropriate ADC will be used. Interface circuitry on the board transmits this value as serial data via optical couplers to the ADC Interface card in the PSCU.

## 2.2 ADC Interface

The ADC Interface card in the PSCU controls the selection and scanning of the remote ADC outputs. It converts the serial data from the remote ADC to parallel data

that is read by the CPU. All signals are optically coupled at both ends. This card can service 8 remote ADCs.

## 2.3 Timer Card

The pulse count generator is the heart of the Timer card. It is implemented by programming a counter in the 9513 IC (5 16-bit counters 40-pin chip) as a down counter fed from another counter programmed as a rate generator. Thus 2 counters are needed for each controlled power supply. The advantage in using a programmable counter chip for generating the required pulses versus a software generated pulse count (which ties up the CPU for the duration of the pulse counting) is the minimal demand on the CPU resource required to program the counter. Counter 5 of the first 9513 chip is programmed as a rate generator of a 20 kHz square wave. Its output is buffered by four differential line drivers that are used to supply the reference clock in each chopper supply. This card also contains 8 bits each of digital inputs and outputs. The outputs are used to encode four count directions DIR[4] and four sync mode values SyncM[4]. Thus two Timer cards are needed for each PSCU. The digital inputs are used to monitor the busy signals from each counter.

## 2.4 GESADC-3

Other analog inputs are connected differentially using shielded twisted pairs to the GESADC-3 in the PSCU. In the differential mode, the minimum gain for the GESADC-3 is 10. This limits the inputs to ±1 volt. Each GESADC-3 has capacity for 16 differential inputs. Therefore two cards are required for each PSCU.

## 2.5 GESINP-2A

Digital signals to the optically coupled GESINP-2A must be 12 volts for a high logic level across 1.6 K$\Omega$ that is referenced to a different ground from the PSCU ground. Zero volt is logical low. Each card has 32 bits of input. Since each supply is alloted 8 bits of digital input, 2 cards are required for each PSCU.

## 2.6 GESOUT-3A

Digital signal outputs from the GESOUT-3A are open collectors with 100 ma sink capability each. The output circuit must supply separate ground (emitter) and load (collector) connected to external supply for each output. Each card is capable of 32 bits of output. Since each supply is alloted 8 bits of digital output, 2 cards are required for each PSCU.

## 3.0 Formats -

## 3.1 Received (Command) Message Formats -

The breakdown of the command message is as follows:

| Bytes | Description |
|-------|-------------|
| byte 0 | Command Code |
| bytes 1,2,3 | CabNum |
| byte 4 | PS #   ( a relative value 0..7) |
| bytes 5..17 | Arguments |
| | (actual # of bytes is command dependent) |

## 3.2 Transmitted (Response) Formats -

Command dependent. Not finalized. Type ASCII.

## 4.0 Command Codes -

| ASCII Code | Hex Code | Function |
|------------|----------|----------|
| A | 0x41 | Ramp current to value |
| B | 0x42 | Set boolean variable |
| C | 0x43 | Reset boolean variable |
| D | 0x44 | Set ramp rate |
| E | 0x45 | Return analog value |
| F | 0x46 | Return boolean value |
| G | 0x47 | Set SyncMode flag |
| H | 0x48 | Reset SyncMode flag |
| I | 0x49 | Return power supply state |
| J | 0x4A | Ramp relative value |
| K | 0x4B | Enable analog monitor |
| L | 0x4C | Disable analog monitor |
| R | 0x52 | Set software DAC value |
| S | 0x53 | Change scale factor |
| U | 0x55 | Set DAC range |
| X | 0x58 | Exit to shell |

Table 1. PSCU Command Codes Supported

The command codes supported by the PSCU are shown in Table 1. This is not meant to be a complete list. However, it should be noted that each additional command supported will require more memory in the PSCU which may result in having to add another memory card. The cost impact ( for 200 PSCU's for the storage ring alone) will have to be carefully considered.

## 4.1 Command Descriptions -

### Ramp To Value (A)

The host computer sends a maximum of 18 bytes ASCII string message to the PSCU. To command a power supply to ramp to an absolute current level, only the first 12 bytes are processed. The bytes are as described in section 3.1. Byte 5 can be a polarity value. If fewer than 18 bytes are issued the command must be terminated by a carriage return.

The command decoder portion of the software will determine the direction and number of pulses required to change the present current setting to the value requested. It then loads the appropriate counters with the required parameter values. The output pulses are fed to an up/down counter that in turn drives a DAC that generates the reference voltage for the regulator loop. The DAC ramps immediately if SyncMode is 0 or waits for a trigger if SyncMode is 1. The input clock frequency to the timer is determined by the Ramp Rate Clock frequency set by the last Set Ramp Rate command (D).

### Set/Reset Boolean Variable (B/C)

These two commands process only the first 6 bytes of the command message. The bitmap argument, byte 5 of the command message, indicates the logic variables affected by the command. Each high bit in the byte identifies the logic variable to activate on a Set command or to de-activate on a Reset command. Eight control outputs can therefore be controlled for each power supply/magnet pair.

The bitmap assignment for SET/RESET command follows:

| | |
|---|---|
| bit 0 | control 0 |
| bit 1 | control 1 |
| bit 2 | control 2 |
| bit 3 | control 3 |
| bit 4 | control 4 |
| bit 5 | control 5 |
| bit 6 | control 6 |
| bit 7 | control 7 |

## Set Ramp Rate (D)

The Ramp Rate argument is a minimum 5-byte value. This value is used to set the initial count in a frequency divider that functions as the rate generator. A maximum of 10 bytes are processed by this command. The valid value for this argument is a minimum of 2 and a maximum of 65535. The PSCU calculates the load parameters, loads them to the appropriate timer and starts the clock. The frequency generated in kHz is given by the relation

$$freq= 800/(2*value).$$

This gives a frequency range from about 6 milliHz to 200kHz. (Note that only 65534 different frequencies are obtainable and in an inverse function.) This frequency output is used by the Ramp To Value commands.

## Return Analog Value (E)

The argument byte 5 of the command message serves as the ID of the variable addressed in the selected power supply/magnet pair. The first six bytes are processed by this command. The assignment of the ID values (subject to change) is as follows:

| Byte 5 value | Addressed variable |
|---|---|
| 0 | Magnet current |
| 1 | Magnet voltage |
| 2 | DAC voltage output |
| 3 | Switcher heat sink temp |
| 4 | Cooling water temp |

## Return Boolean Value (F)

In the RETURN BOOLEAN command the returned byte is the bitmap of status of all the assigned logic functions for the addressed magnet/power supply unit. The first five bytes are processed by this command. The bitmap assignment of the returned status byte follows the same pattern used in the SET/RESET command.

| | |
|---|---|
| bit 0 | status 0 |
| bit 1 | status 1 |
| bit 2 | status 2 |
| bit 3 | status 3 |
| bit 4 | status 4 |
| bit 5 | status 5 |
| bit 6 | status 6 |
| bit 7 | status 7 |

### Set Sync Mode (G)

No additional argument byte beyond the power supply number is required for this command. Only the first five bytes are processed. Upon execution of command the sync enable bit SyncM[] of the addressed power supply is raised. The timer will not generate the required pulses unless a trigger pulse is received by the timer card. The addressed supply will remain in this mode until a Reset Sync Mode (H) command is received by it.

### Reset Sync Mode (H)

No additional argument byte beyond the power supply number is required for this command. Only the first five bytes are processed. Upon execution of command the sync enabled bit in the timer card is reset. The PSCU will generate the required pulses immediately upon receiving a ramp command. This is the default mode for all power supplies.

### Return Power Supply State (I)

Only the first five bytes are processed by this command. The information returned from the addressed power supply database are:

> SyncMode status if on
> Reference DAC range
> Reference DAC value
> Magnet current value
> Status byte
> Control byte
> Reference DAC voltage out
> Magnet voltage
> Heat sink temperature
> Cooling water temperature

### Ramp Relative Value (J)

The first 12 bytes are processed by this command. Bytes 5 to 11 represent the number of pulses generated by the timer circuit if the resulting absolute DAC value is within the DAC range for the power supply. If the argument value will cause an over or under-range, only that number of pulses sufficient to cause positive or negative full scale will be generated. The DAC value will be updated to reflect the present state.

### Monitor Analog On (K)

Only the first six bytes are processed by this command. Byte 5 of the command string is the pointer ID of the variable to monitor on the console. The alarm monitoring continues while analog monitoring is on. The ID assignments are as follows:

| ID # | Variable |
|------|----------|
| 0 | Magnet current value |
| 1 | Magnet voltage |
| 2 | DAC voltage |
| 3 | Heat sink temperature |
| 4 | Cooling water temperature |

## Monitor Analog Off (L)

Stops the display of the analog channel. Only the first five bytes are processed.

## Set Software DAC Value (R)

First 12 bytes are processed by this command. Bytes 5..11 specify the new DAC value without generating pulses to the DAC. This command is used to reconcile the DAC hardware input and the value used in the PSCU software.

## Change Scale Factor (S)

This command uses the most bytes. A maximum of 18 bytes may be processed. The additional bytes are required so that the E-format of number specification can be accomodated. Bytes 5..17 specify the new scale factor to use for the addressed power supply. The old scale factor is also displayed.

## Set DAC Range (U)

The first six bytes are processed by this command. Byte 5 specifies the new range code as follows:

| Range Code | DAC Range |
|------------|-----------|
| Unipolar | |
| 0 | 0..4095 |
| 1 | 0..16383 |
| 2 | 0..65535 |
| 3 | 0..262143 |
| Bipolar | |
| 4 | -2048..2047 |
| 5 | -8192..8191 |
| 6 | -32768..32767 |
| 7 | -131072..131071 |

Exit To Shell (X)

    Exits to shell and restarts the program.

## 5.0 Control Logic -

    The PSCU maintains a number of buffers and flags that allow it to hold messages, data and status information during its operation. The PSCU receives messages from the host via the LAN. It decodes the message and executes the command. If no command from the host is active, the PSCU executes the software loop of monitoring for host messages, continual update of its power supply statuses and data buffers, and alarm checking. Thus the data in each power supply database is virtually live.

    The alarm status received by the PSCU does not automatically cause the PSCU to generate a shut down signal to the interlock chain (though it can be used to do so). The interlock chain is a direct hardware implementation, therefore it does not require any computer or other external logic control in order to function. The eight digital outputs per power supply pair can be used to control or initiate external events.

    Below is a C-language pseudo code for the PSCU logic that describes in a general way the main idea of its operation after some requisite variable and port initializations.

```
main()
{

Do
{      Read Status Inputs;
       Read Analog Inputs;
       Update PSCU Database;
       Check Alarm Status;
       if (Alarm Condition)
       Report Alarm to Host;
} while forever  /* endless do--while loop */

}
```

    The handler that services the interrupt message from the host determines what command is received and what power supply is addressed. Other routines in the software decompose the command string further into the various arguments required for each specific command.